IN THE CLAIMS

1. (Previously Presented) A cache memory device, comprising:

a plurality of memory cells adapted for caching array data traffic separate from other data traffic; and

at least one register adapted for storing access information for accessing at least one array cached in the plurality of memory cells.

2. (Original) The cache memory device of claim 1, wherein the access information includes an array ID for identifying an array of memory cells within the plurality of memory cells.

3. (Original) The cache memory device of claim 1, wherein the access information includes array information used for offset and boundary calculations.

4. (Original) The cache memory device of claim 3, wherein the array information used for offset and boundary calculations includes array height information and array width information.

5. (Original) The cache memory device of claim 4, wherein the array information used for offset and boundary calculations further includes array stride information.

6. (Original) The cache memory device of claim 1, wherein each of the at least one register contains information that corresponds to an array data structure within a main memory.

7. (Previously Presented) A cache memory device, comprising:

a plurality of memory cells adapted for caching array data traffic separate from other data traffic; and

at least one register adapted for storing access information for accessing at least one array cached in the plurality of memory cells, wherein the access information includes:

an array ID for identifying an array of memory cells within the plurality of memory cells;

array height information;

array width information; and

array stride information.

8. (Previously Presented) The cache memory device of claim 7, wherein the array height information, the array width information and the array stride information are adapted for offset and boundary calculations to access the array of memory cells.

9. (Previously Presented) The cache memory device of claim 7, wherein the access information contained within the at least one register corresponds to one or more array data structures within a main memory.

10. (Previously Presented) The cache memory device of claim 1, wherein the access information includes:

a base address for identifying a contiguous region of memory storage within a main memory; and

array information for boundary and memory offset calculations to access the array of memory cells.

11. (Original) An electronic system, comprising:

an array cache memory device adapted for caching array data; and

a boundary policy enforcement and index mapping unit coupled to the cache memory device.

12. (Original) The electronic system of claim 11, wherein the boundary policy enforcement and index mapping unit is adapted for storing array attributes.

13. (Original) The electronic system of claim 11, wherein the boundary policy enforcement and index mapping unit is adapted for using the array attributes to identify known array limits and to confine a range of indices to conform within the known limits.

14. (Original) The electronic system of claim 13, wherein the boundary policy enforcement and index mapping unit is adapted for converting confined multi-dimensional indices into one-dimensional offsets.

15. (Original) The electronic system of claim 11, wherein the array cache memory device includes:

a plurality of memory cells; and

at least one register adapted for storing access information for accessing at least one array stored in the plurality of memory cells.

16. (Original) The electronic system of claim 15, wherein the access information includes an array ID for identifying an array of memory cells within the plurality of memory cells.

17. (Original) The electronic system of claim 15, wherein the access information includes array information used for offset and boundary calculations.

18. (Original) The electronic system of claim 17, wherein the array information used for offset and boundary calculations includes array height information and array width information.

19. (Original) The electronic system of claim 18, wherein the array information used for offset and boundary calculations further includes array stride information.

20. (Original) The electronic system of claim 15, wherein each of the at least one register contains information that corresponds to an array data structure within a main memory.

21. (Original) The electronic system of claim 11, further comprising:

a main memory coupled to the array cache memory device;

a general cache memory device coupled to the main memory and adapted for caching regular data; and

a processor coupled to and adapted for communication with the main memory, the general cache memory device and the array cache memory device.

22. (Original) The electronic system of claim 11, further comprising a memory arbitrator coupled to and adapted for communication with the main memory, the general cache memory device and the array cache memory device.

23. (Original) An electronic system, comprising:

a main memory;

a dynamic array cache memory device coupled to the main memory and adapted for caching array data;

a general cache memory device coupled to the main memory and adapted for caching regular data; and

a processor coupled to and adapted for communication with the main memory, the general cache memory device, and the dynamic array cache memory device.

24. (Original) The electronic system of claim 23, further comprising a memory arbitrator coupled to and adapted for communication with the main memory, the general cache memory device, and the dynamic array cache memory device.

25. (Original) The electronic system of claim 23, wherein:

the dynamic array cache memory device includes at least one register adapted for storing array access information;

the main memory includes at least one allocated memory region containing array elements; and

the array access information stored in each register corresponds to the at least one allocated memory region.

26. (Original) The electronic system of claim 25, wherein the array access information includes information for boundary and memory offset calculations.

27. (Original) The electronic system of claim 26, wherein the array access information includes array height information and array width information.

28. (Original) The electronic system of claim 27, wherein the array access information used for boundary and memory offset calculations for the array of memory cells further includes array stride information.

29. (Original) An electronic system, comprising:

a main memory;

a dynamic array cache memory device coupled to the main memory and adapted for caching array data, the dynamic array cache memory device including:

a plurality of memory cells; and

at least one register adapted for storing array access information, including:

a base address for identifying a contiguous region of memory storage within the main memory; and

array information for boundary and memory offset calculations;

a general cache memory device coupled to the main memory and adapted for caching regular data; and

a processor coupled to and adapted for communication with the main memory, the general cache memory device, and the dynamic array cache memory device.

30. (Original) The electronic system of claim 29, wherein the array information for boundary and memory offset calculations includes array height information and array width information.

31. (Original) The electronic system of claim 30, wherein the array information for boundary and memory offset calculations further includes array stride information.

32. (Currently Amended) The electronic system of claim 30, further including a main memory arbitrator for controlling communication between the main memory and the general cache memory device, and between the main memory and the dynamic array cache memory device.

33. (Original) The electronic system of claim 30, wherein:

the main memory includes at least one contiguous memory region; and

the array access information stored in each of the at least one register corresponds to the at least one contiguous memory region.

34. (Original) The electronic system of claim 30, further including a dynamic array code residing in the main memory and adapted for being operated on by the processor, wherein the dynamic array code is adapted for separating array access traffic from regular data traffic.

35. (Original) The electronic system of claim 34, wherein:

the dynamic array code provides at least one region of main memory; and

the array access information stored in each register corresponds to the at least one region of main memory.

36. (Currently Amended) The electronic system of claim 30, further comprising a boundary policy enforcement and index mapping unit coupled to the processor and the dynamic array cache memory.

37. (Original) The electronic system of claim 36, wherein the boundary policy enforcement and index mapping unit is adapted for storing array attributes.

38. (Original) The electronic system of claim 36, wherein the boundary policy enforcement and index mapping unit is adapted for using the array attributes to identify known array limits and to confine a range of indices to conform within the known limits.

39. (Original) The electronic system of claim 38, wherein the boundary policy enforcement and index mapping unit is adapted for converting confined multi-dimensional indices into one-dimensional offsets.

40. (Previously Presented) A cached dynamic array (CDA) system, comprising:

a main memory;

a dynamic array cache memory coupled to the main memory and adapted for caching array data;

a data cache memory coupled to the main memory and adapted for caching regular data;

an instruction cache memory coupled to the main memory and adapted for caching processor instructions;

a processor coupled to and in communication with the general cache memory, the dynamic array cache memory, and the instruction cache memory; and

a computer readable medium encoded with a compiled software program capable of being executed by the processor such that the processor is adapted for:

allocating a region of allocated memory for storing array elements;

allocating and initializing an array descriptor register, and obtaining a corresponding array handle to the array descriptor register;

overlaying an access structure on the region of memory by associating a pointer to the memory region with the array descriptor register; and

accessing array elements using indices and the array handle.

41. (Original) The CDA system of claim 40, wherein the software program includes a declaration that provides a pointer to a corresponding allocated memory array, an array identifier containing the array handle value, and array information with which boundary and memory offset calculations are made.

42. (Original) The CDA system of claim 40, wherein the software program includes a declaration that provides a pointer to a corresponding allocated memory array, an array identifier containing the handle value, array height information, array width information and array stride information.

43. (Original) A cached dynamic array (CDA) system, comprising:

a main memory;

a dynamic array cache memory coupled to the main memory and adapted for caching array data;

a general cache memory coupled to the main memory and adapted for caching regular data;

a processor coupled to and in communication with the main memory, the general cache memory, and the dynamic array cache memory; and

a computer readable medium encoded with a compiled software program capable of being executed on the processor such that the processor is adapted for accessing dynamic array elements and reading memory data on a hardware path separate from a path used for regular data traffic.

44. (Original) The CDA system of claim 43, wherein the processor executes instructions that use a handle value to select an array, and indices to access the array element.

45. (Original) The CDA system of claim 43, wherein the dynamic array cache memory includes:

a plurality of memory cells; and

at least one register adapted for storing access information for accessing at least one array stored in the plurality of memory cells.

46. (Original) The electronic system of claim 45, wherein the access information includes an array ID for identifying an array of memory cells within the plurality of memory cells.

47. (Original) The electronic system of claim 45, wherein the access information includes array information used for offset and boundary calculations.

48. (Original) The electronic system of claim 47, wherein the array information used for offset and boundary calculations includes array height information and array width information.

49. (Original) The electronic system of claim 48, wherein the array information used for offset and boundary calculations further includes array stride information.

50. (Original) The electronic system of claim 45, wherein each of the at least one register contains information that corresponds to an array data structure within a main memory.

51. (Original) A method of caching declared dynamic arrays, comprising:

separating dynamic array accesses from regular data traffic; and

caching the dynamic array accesses separately from the regular data traffic.

52. (Original) The method of claim 51, wherein separating dynamic array accesses from regular traffic includes providing instructions to a processor such that the processor is adapted for accessing array data using a hardware path separate from a path for normal processor data access.

53. (Original) The method of claim 51, wherein caching the dynamic array accesses separately from the regular data traffic includes:

storing array attribute information in a plurality of hardware registers;

accessing the array attribute information using an array handle; and

generating a real memory address using array indices with the attribute information; and

forwarding the real memory address to a dynamic array cache as a memory access request.

54. (Original) The method of claim 53, further comprising calculating the real memory address for the dynamic array cache based on array boundary policies, array height, array width, array stride, array element size, and array indices.

55. (Original) A method of prefetching array data in an electronic system, comprising:

separating each dynamic array access from regular data traffic;

caching each dynamic array access into a dynamic array cache; and

speculatively loading data based on anticipated dynamic array accesses into the dynamic array cache.

56. (Original) The method of claim 55, wherein speculatively loading data includes determining that access has moved to a predefined point in currently accessed data, and checking that the data to be speculatively loaded is not already loaded into the cache.

57. (Original) The method of claim 56, wherein the data to be speculatively loaded is selected using array attribute information.

58. (Original) The method of claim 55, further comprising reducing main-memory bandwidth by using a copy-back cache design.

59. (Original) The method of claim 55, wherein separating each dynamic array access from regular traffic includes providing processor instructions for accessing array elements using a hardware data path that is separate from a data path used by other processor instructions.

60. (Original) The method of claim 55, wherein caching each dynamic array access includes caching data at a memory address that is generated from each dynamic array access.

61. (Original) The method of claim 60, wherein caching data at a memory address includes:

generating a memory address from each dynamic array access by applying an array handle to obtain array attributes; and

calculating the memory address using the attributes and array indices.

62. (Original) The method of claim 61, further comprising storing the array attributes in memory that is addressed by the array handle and is separate from the main memory.

63. (Original) The method of claim 62 further comprising representing the array handle with a number of bits that is fewer than a number of bits used to represent a main memory address.

64. (Original) The method of claim 60, further comprising calculating the memory address for a CDA cache based on array height, array width, array stride and array indices.